Image Retrieval with Local and Spatial Queries

Baback Moghaddam, Henning Biermann and Dimitris Margaritis
Mitsubishi Electric Research Laboratory, USA
New York University, USA
Carnegie Mellon University, USA

ABSTRACT

To date most "content-based image retrieval" (CBIR) techniques rely on *global* attributes such as color or texture histograms which unfortunately ignore the spatial composition of the image. In this paper, we present an alternative image retrieval system based on the principle that it is the *user* who is most qualified to specify the query "content" and *not* the computer. With our system, the user can select multiple "regions-of-interest" and can specify the relevance of their spatial layout in the retrieval process. We also derive similarity bounds on histogram distances and use them to prune the database search. This experimental system was found to be superior to global indexing techniques as measured by statistical sampling of multiple users' "satisfaction" ratings.

1 Introduction

The use of locality and spatial configuration in image retrieval is exemplified by recent work on automatic blob segmentation and description, as in Howe's "percentile blob" technique [4] and the "Blobworld" segmentation system of Carson *et al.* [1]. We propose a system which differs from the above in one key respect: there are no pre-segmented regions. Instead, the user defines "blobs" or "regions-of-interest" (ROIs) directly on a query image (and implicitly their relative spatial configuration) in order to define the "content" to be retrieved. The disadvantage of this scheme, however, is that region-matching and subsequent database indexing must be done in an online fashion and moreover in "interactive-time" to be tolerated by the user. Aggressive search pruning and use of similarity bounds does, however, alleviate this problem.

2 Representation and Similarity

An image retrieval system is completely defined by two basic specifications: representation (of features) and a corresponding similarity metric (for comparison of features or their distributions). Our system divides the image into an array of 16-by-16 pixel blocks wherein each pixel yields a LUV color coordinate and three texture measurements; edge strength: $\log(G_x^2 + G_y^2)$, Laplacian: $G_{xx} + G_{yy}$ and edge orientation: $\arg(G_x, G_y)$, where G_x and G_{xx} are the 1st and 2nd derivatives of a Gaussian filter with specified scale σ. In our experiments, two separate scale parameters were used: $\sigma = 1$ and $\sigma = 2$, yielding two sets of ("independent" or at least uncorrelated) texture measurements. We also implemented and tested 3 different histogram similarity measures for our data representation: Histogram Intersection [6], Chi-squared statistic [5] and Bhattacharyya distance [3].

3 Region Matching

Block histograms represent *local* color and texture and due to the additive property of histograms, can be easily combined (summed) to form densities for larger image blocks, including the entire image at which point they become identical to global histograms. When the user specifies a region of interest, its underlying block histograms are "pooled" to represent a "meta-block" histogram as illustrated in Figure 1. A region is then used to index into the database, where an online search for the best matching region (of the same size) is conducted using the aforementioned similarity metrics. Multiple region queries are processed in parallel and the best region match scores are then combined (usually by summation) to determine the final visual similarity ranking.

To speed up the online search, the entire database is first pruned to find a small subset (typically 5-10%) of "compatible" images with fast *global* histogram

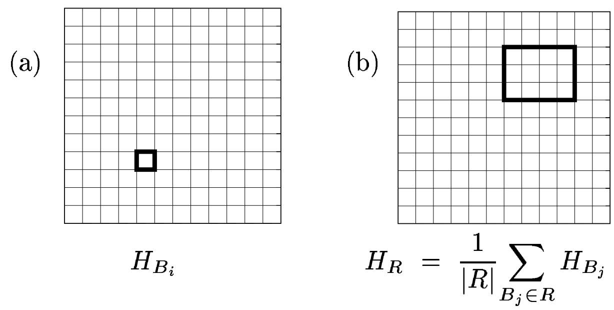


Figure 1: (a) An image block B_i with corresponding histogram H_{B_i} (b) A region R composed of individual blocks B_j and its "pooled" histogram H_R

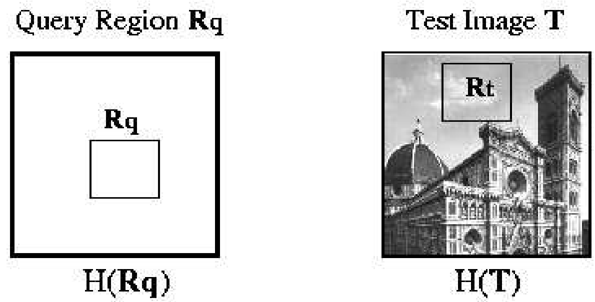


Figure 2: A query R_q and its candidate best matching region R_t in a database image T.

indexing using a *branch and bound* algorithm. The key observation is that we can find lower bounds for the query distances using pooled-ROI and global histogram comparisons. Consider a query region histogram H_R^q and a database test image with candidate regions H_R^t and a global histogram H^t, as shown in Figure 2. It can be easily shown that with a distance metric based on histogram intersection (distance denoted here by $\perp$) one can compute an exact lower bound on the region-to-region distance

$$H_R^q \perp H_R^t \quad \geq \quad H_R^q \perp H^t \quad \ldots \quad \text{for } \textbf{any } H_R^t \quad (1)$$

Since similarity (inverse distance) is simply a count of the number of pixels in common, the similarity between the query and the global test image can never be smaller than the similarity between the query region and a corresponding subregion in the test image. For other metrics such as Chi-squared and Bhattacharyya, exact lower bounds are difficult to compute. Nevertheless, approximate lower bounds similar to Equation 1 can still be of practical use. Once the database is sufficiently pruned, we search for all combinations of regions in the target image in order to find the best matching regions (this is potentially very slow but the user is expected to specify only few ROIs).

4 Spatial Constraints

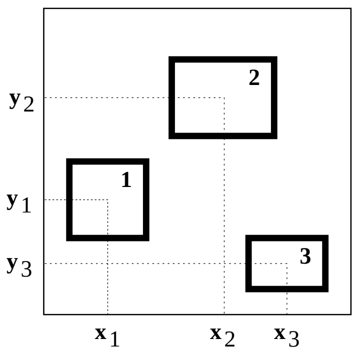
In addition to querying by visual similarity, the user also has the option of specifying whether the selected regions should maintain their respective spatial configuration in the retrieved matches. We used a simple formulation based on the consistency of binary relations on the centroid coordinates of the regions, as illustrated in Figure 3. Given the user-defined query Q, consisting of n regions, its spatial configuration similarity to a candidate configuration T (with corresponding best matching regions) is

$$S = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} f(x_i^t - x_j^t)\, s(x_i^q - x_j^q) + f(y_i^t - y_j^t)\, s(y_i^q - y_j^q) \qquad (2)$$

where x_i^q and x_i^t are the region centroid coordinates of the query Q and candidate T, respectively. The function f is a bipolar sigmoid (hyperbolic tangent) and its product with the sign function $s()$ will essentially result in a "fuzzy" or "soft" count of the total number of satisfied constraints (in the set of binary relations) between Q and T. The scale parameter of the sigmoid function can be adjusted to specify how strictly a binary constraint is imposed (in the limit f can be made into a sign function as well). This formulation is related to the technique of "2D strings" proposed for iconic indexing by Chang et al. [2]

5 Experimental Results

Due to the multiplicity of "content" in a single image, the only sensible performance measure is one that quantifies the user's overall "satisfaction" with the retrieved matches. Our experimental design was simple: 31 naive users were instructed in the basic operations of the multiple ROI query interface and asked to perform a minimum of 20 different queries on various databases. Each user-defined region-based retrieval was presented with the resulting *global* search with the same query image. The user then had to select (forced choice) which set of retrievals (local or global) captured the "essence" of their intended query content. From this sample of more than 600 (mostly



$$x_1 < x_2 \ , \ x_1 < x_3 \ , \ x_2 < x_3$$

$$y_1 < y_2 \ , \ \overline{y_1 < y_3} \ , \ \overline{y_2 < y_3}$$

Figure 3: Three regions and the complete set of binary relationships corresponding to their spatial configuration.

independent) selections the average percentage of acceptable local first-rank matches was found to be 88% ($F_{1,30} = 14.8$, $p \leq 0.05$), indicating that local searches were in fact favored over global ones.

Figure 4 shows an example query in our browser, running on a database of GIS Orthophoto Imagery of the state of Massachusetts (available at http://ortho.mit.edu). The smaller window in the lower left allows the user to graphically define and edit (in this case) three regions corresponding roughly to "dense urban row housing", "water" and "factory" region types (note that these "classes" are entirely user-defined). The user can either retrieve images which respect the spatial configuration of the query, or alternatively, disable spatial scoring to simply retrieve images containing similar types of regions.

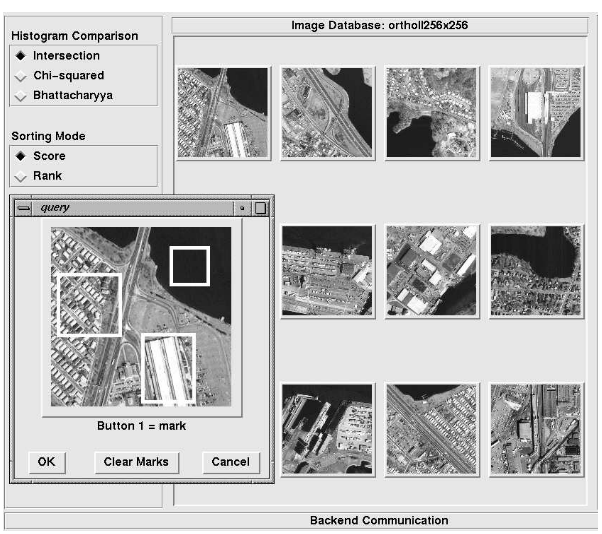


Figure 4: An example of a multiple ROI query with a database of B&W aerial imagery.

6 Discussion

Currently the online search for individual regions is computationally intensive and more sophisticated pruning strategies should be implemented in order to avoid searching every region of every image in the database. Global histogram indexing with the *branch and bound* method is effective in pruning the candidate set. Another speed-up possibility is to immediately reject candidate images based on partial spatial configurations (*e.g.,* if the best match for region 2 is already on the wrong side of region 1, reject the current image). We believe that our system should be especially useful for domain-specific databases such as the GIS aerial imagery shown in Figure 4 where global descriptors and lack of spatial configuration form a rather poor representation of content.

References

[1] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Region-based image querying. In *Proc. IEEE Workshop on Content-Based Access of Image and Video Libraries*, June 1997.

[2] S. Chang, Q. Shi, and S. Yan. Iconic indexing using 2-D strings. *IEEE Trans. on Pattern Analysis & Machine Intelligence*, 9(3):413–428, May 1987.

[3] K. Fukunaga. *Introduction to Statistical Pattern Recognition.* Academic Press, 1971.

[4] N. Howe. Percentile blobs for image similarity. In *Proc. IEEE Workshop on Content-Based Access of Image and Video Libraries*, June 1998.

[5] A. Papoulis. *Probability, Random Variables, and Stochastic Processes.* McGraw Hill, 1991.

[6] M. Swain and D. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.